Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

Claims 1-7 (Cancelled)

8. (Previously Presented) A high-performance specification resolution method for use in detecting attacks against computer systems, comprising:

a) receiving audit conditions to be detected using non-limiting specification formulas expressing fraudulent entry or attack patterns or abnormal operations, to be verified by examining records of a log file of a computer system;

b) expanding said formulas into subformulas for each record;

c) scanning and generating, for each expanded formula, Horn clauses to resolve in order to detect whether or not the formula is valid for each record, the Horn clauses expressing implications resolvent of the subformulas for each record scanned in positive clauses having a positive literal and in negative clauses having at least one negative literal;

d) storing the positive clauses in a stack of subformulas,

storing, in a table of clauses, a representation of the negative clauses and the positive clauses;

storing, in a table of counters, a number of negative literals in each negative clause;

e) resolving the table of clauses based on each positive clause so as to generate either an output file or an action of the computer system;

f) iterating steps b) through e) until the scanning of all the records in the log file is complete.

9. (Currently Amended) The method according to claim 8, wherein temporal logic is used for ~~the~~ formulation of the high-performance specification.

10. (Previously Presented) The method according to claim 8, wherein the table of clauses is a matrix indexed in columns by subscripts of the formulas appearing in the negative clauses, and indexed in lines by subscripts of the formulas appearing in the positive clauses.

11. (Previously Presented) The method according to claim 8, wherein the table of clauses is a sparse matrix, the columns being represented by chained lists.

12. (Previously Presented) The method according to claim 8, further comprising optimizing the expansion of the formulas using a hash table to ensure that a formula is not expanded more than once in each record.

13. (Previously Presented) The method according to claim 9, further comprising optimizing the expansion of the formulas using a hash table to ensure that a formula is not expanded more than once in each record.

14. (Previously Presented) The method according to claim 8, wherein the log file is scanned only once from beginning to end.

15. (Previously Presented) A computer system comprising:

storage means; and

a processor, coupled to the storage means, for executing programs implementing a high performance resolution method for detecting attacks against the system wherein the processor operates to:

a) receive audit conditions to be detected using non-limiting specification formulas expressing fraudulent entry or attack patterns or abnormal operations, to be verified by examining the records of a log file;

b) expand said formulas into subformulas for each record;

c) scan and generate, for each expanded formula, Horn clauses to resolve in order to detect whether or not the formula is valid for each record, the Horn clauses expressing implications resolvent of the subformulas for each record scanned in positive clauses having a positive literal, and in negative clauses having at least one negative literal;

d) store the positive clauses in a stack of subformulas,

store, in a table of clauses, a representation of the negative clauses and the positive clauses,

store, in a table of counters, a number of negative literals in each negative clause; and

e) resolve the table of clauses based on each positive clause, so as to generate either an output file or an action of the computer system.

16. (Currently Amended) The computer system according to claim 15 wherein temporal logic is used for formulation of the high-performance resolution method~~specification~~.

17. (Previously Presented) The computer system according to claim 15, wherein the table of clauses is a matrix indexed in columns by subscripts of the formulas appearing in the negative clauses, and in lines by subscripts of the formulas appearing in the positive clauses.

18. (Previously Presented) The computer system according to claim 15, wherein the table is a sparse matrix, the columns being represented by chained lists.

19. (Previously Presented) The computer system according to claim 15 including a hash table to ensure that a formula is not expanded more than once in each record.

20. (Previously Presented) The computer system according to claim 16 including a hash table to ensure that a formula is not expanded more than once in each record.

21. (Previously Presented) The computer system according to claim 15 including means for scanning the log file only once from beginning to end.

22. (Previously Presented) The computer system according to claim 15, wherein the programs executed by the processor include:

an adaptor module for translating information from the log file;

an interpreter module for receiving the information from the adapter, receiving the specification formulas, expanding the specification formulas, and filling in the table of clauses, table of counters and the stack of subformulas stored in the storage means; and

a clause processing module for resolving the Horn clauses using the information from the table of clauses, the table of counters and the stack of worked subformulas, the clause processor generating the output file or generating the action.